What is claimed is:

1. A method of reading desired data from a remote storage device that receives chunks of data from an other storage device, comprising:

    determining if the desired data is part of a chunk of data committed by the other

5   storage device;

    if the desired data is not part of a chunk of data committed by the other storage device, reading the desired data from a corresponding standard logical device; and

    if the desired data is part of a chunk of data committed by the other storage device, transferring the desired data to the standard logical device and obtaining the

10   desired data from one of: the standard logical device after transferring the desired data thereto and the chunk of data committed by the other storage device.

2. A method, according to claim 1, further comprising:

    if the desired data is part of a chunk of data committed by the other storage device, locking a corresponding slot of a cache only virtual device that points to the

15   desired data.

3. A method, according to claim 2, further comprising:

    after locking the corresponding slot, redetermining if the desired data is part of a chunk of data committed by the other storage device.

4. A method, according to claim 3, further comprising:

if the result of redetermining indicates that the desired data is not part of a chunk of data committed by the other storage device, unlocking the corresponding slot and reading the desired data from a corresponding standard logical device.

5   5. A method, according to claim 3, further comprising:

locking a corresponding slot of the standard logical device;

merging corresponding cache slots; and

causing the corresponding slot of the cache only virtual device to point to the standard logical device.

10   6. A method for a host coupled to a remote storage device to read desired data transmitted by a local storage device, comprising:

identifying a most recent and consistent set of data containing the desired data; and

obtaining the desired data from the most recent and consistent set of data.

15   7. A method, according to claim 6, wherein the set of data is a chunk of data committed by the local storage device.

8. A method, according to claim 7, wherein the chunk of data is assigned a sequence number that is less than a sequence number for other chunks of data containing other data that an other host computer started to write after starting to write the desired data.

/

60

9. A method, according to claim 8, further comprising:

determining if multiple tracks of the remote storage device are being read;

if multiple tracks are being read, determining a first current sequence number

prior to reading the tracks, reading the tracks, and determining a second current sequence

5   number; and

if the first current sequence number does not equal the second current sequence

number, rereading the tracks.

10. A method, according to claim 8, further comprising:

determining if multiple tracks of the remote storage device are being read;

10   if multiple tracks are being read, determining a first current sequence number

prior to reading the tracks, reading the tracks, and determining a second current sequence

number; and

if the first current sequence number does not equal the second current sequence

number, returning an error.

15

11. Computer software that reads desired data from a remote storage device that receives

chunks of data from an other storage device, the software comprising:

executable code that determines if the desired data is part of a chunk of data

committed by the other storage device;

5   executable code that reads the desired data from a corresponding standard logical

device if the desired data is not part of a chunk of data committed by the other storage

device; and

executable code that transfers the desired data to the standard logical device and

obtains the desired data from one of: the standard logical device after transferring the

10   desired data thereto and the chunk of data committed by the other storage device if the

desired data is part of a chunk of data committed by the other storage device.

12. Computer software, according to claim 11, further comprising:

executable code that locks a corresponding slot of a cache only virtual device that

points to the desired data if the desired data is part of a chunk of data committed by the

15   other storage device.

13. Computer software, according to claim 12, further comprising:

executable code that redetermines if the desired data is part of a chunk of data

committed by the other storage device after locking the corresponding slot.

62

14. Computer software, according to claim 13, further comprising:

      executable code that unlocks the corresponding slot and reads the desired data from a corresponding standard logical device if the result of redetermining indicates that the desired data is not part of a chunk of data committed by the other storage device.

5   15. Computer software, according to claim 13, further comprising:

      executable code that locks a corresponding slot of the standard logical device;

      executable code that merges corresponding cache slots; and

      executable code that causes the corresponding slot of the cache only virtual device to point to the standard logical device.

10   16. Computer software for a host coupled to a remote storage device to read desired data transmitted by a local storage device, comprising:

      executable code that identifies a most recent and consistent set of data containing the desired data; and

      executable code that obtains the desired data from the most recent and consistent

15   set of data.

17. Computer software, according to claim 16, wherein the set of data is a chunk of data committed by the local storage device.

18. Computer software, according to claim 17, wherein the chunk of data is assigned a sequence number that is less than a sequence number for other chunks of data containing other data that an other host computer started to write after starting to write the desired data.

5   19. Computer software, according to claim 18, further comprising:

executable code that determines if multiple tracks of the remote storage device are being read;

executable code that determines a first current sequence number prior to reading the tracks, reads the tracks, and determines a second current sequence number if multiple

10   tracks are being read; and

executable code that rereads the tracks if the first current sequence number does not equal the second current sequence number.

20. Computer software, according to claim 18, further comprising:

executable code that determines if multiple tracks of the remote storage device are

15   being read;

executable code that determines a first current sequence number prior to reading the tracks, reads the tracks, and determines a second current sequence number if multiple tracks are being read; and

executable code that returns an error if the first current sequence number does not

20   equal the second current sequence number.